

What is claimed is:

1. A method of generating kernel audit data comprising:
storing system call parameters or data the parameters point to at the
beginning of a system call; and
5 triggering data delivery at the end of the system call path and generating
an audit record and depositing the audit record in a circular buffer.
2. The method of claim 1, wherein each system call that accesses
files, storing related file information.
- 10 3. The method of claim 2, wherein related file information includes
file owner or group and the file information is stored before any modifications
occur that might affect the file information.
- 15 4. The method of claim 1, wherein system call parameters that
include path name parameters are stored with full path name information.
5. The method of claim 1, wherein the audit record is a tokenized
audit record.
- 20 6. The method of claim 1, further comprising reading audit records
from the circular buffer.
7. The method of claim 6, wherein the reading is triggered using a
25 device read call.
8. The method of claim 1, comprising maintaining system wide
configuration related data structures and setting selection masks based on such
structures.

9. The method of claim 1, comprising collecting data in the system call path and formatting the collected data into an audit record.

5 10. The method of claim 9, wherein the collected data is a token stream.

11. The method of claim 1, comprising if the circular buffer is full, then either reading some of the audit records from the circular buffer or dropping
10 new records until space becomes available in the circular buffer.

12. The method of claim 4, comprising maintaining root and current directions while threads are in the middle of system call processing.

15 13. The method of claim 9, comprising selecting which data to collect before said collecting step.

14. The method of claim 13, wherein said selecting step can be based on process, user, group, filename information and/or time intervals.
20

15. The method of claim 1, further comprising detecting hard link accesses to a critical file.

16. The method of claim 15, comprising maintaining a critical file list
25 for monitoring hard links.

17. The method of claim 5, wherein the tokens are either primitive or composed.

09037942801

18. The method of claim 13, wherein said selecting step can be based on the outcome of system calls including pass, failure or both.

19. The method of claim 1, further comprising presenting deposited
5 data to a user space via a device driver in the kernel.

20. The method of claim 13, comprising configuring which system calls are audited by making ioctl() (control) calls on a device driver.

10 21. The method of claim 1, comprising enabling the generation of audit data when a device driver is opened for read, and halting data generation when the device driver is closed.

09967942-111601